Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer system configured to:

A) provide ~~at least one task queue~~ a plurality of task queues, each having a top end and a bottom end and in which can be stored and from which can be retrieved task identifiers, which identify tasks to be performed; and

B) for each provided task queue, employ a separate execution thread associated therewith to:

i) select repeatedly a current access mode from one of a LIFO access mode and a FIFO access mode in accordance with a mode-selection criterion; and

ii) perform dynamically identified tasks, wherein each said dynamically identified task is a garbage-collection task for performing, for a given object associated with that task, processing that includes identifying in the given object references to other objects, and thereby identifying the tasks of performing similar processing for those other objects, by repeatedly:

a) popping a task identifier from one of the top end or the bottom end of that task queue in order to access that task queue in a LIFO access mode or a FIFO access mode in accordance with the current access mode;

b) so performing the task thereby identified as, in at least some instances, to find one or more further tasks to be performed; and

c) pushing onto that task queue task identifiers that identify any tasks thus found.

2. (Previously Presented) A computer system as defined in claim 1 wherein pushing occurs at , the bottom end of each provided task queue, popping in accordance with the FIFO access mode occurs at the top end of each provided queue, and popping in accordance with the LIFO access mode occurs at the bottom end of each provided task queue.

3. (Previously Presented) A computer system as defined in claim 1 wherein queue accesses in each provided task queue are circular.

4.-5. (Canceled).

6. (Currently Amended) A computer system as defined in claim ~~5~~ 1 wherein the task identifiers are identifiers of the objects associated with tasks that the task identifiers identify.

7. (Original) A computer system as defined in claim 6 wherein the task identifiers are pointers to the objects associated with the tasks that the task identifiers identify.

8. (Currently Amended) A computer system as defined in claim [[4]] 1 wherein, in at least some instances, an execution thread associated with a task queue that is empty:

A) pops a task identifier from a task queue other than the one with which it is associated;

B) so performs the task thereby identified as, in at least some instances, to find one or more further tasks to be performed; and

C) pushes onto the task queue associated with it task identifiers that identify any tasks thus found.

9. (Original) A computer system as defined in claim 8 wherein each said dynamically identified task is the garbage-collection task of performing, for a

given object associated with that task, processing that includes identifying in the given object references to other objects and thereby identifying the tasks of performing similar processing for those other objects.

10.-14. (Canceled).

15. (Currently Amended) For performing dynamically identified tasks, a method comprising employing a computer system to:

A) provide ~~at least one task queue~~ a plurality of task queues, each having a top end and a bottom end and in which can be stored and from which can be retrieved task identifiers, which identify tasks to be performed; and

B) for each provided task queue, employ a separate execution thread associated therewith to:

i) select repeatedly a current access mode from one of a LIFO access mode and a FIFO access mode in accordance with a mode-selection criterion; and

ii) perform dynamically identified tasks, wherein each said dynamically identified task is the garbage-collection task of performing, for a given object associated with that task, processing that includes identifying in the given object references to other objects and thereby identifying the tasks of performing similar processing for those other objects, by repeatedly;

a) popping a task identifier from one of the top end or the bottom end of that task queue in order to access that task queue in a LIFO access mode or a FIFO access mode in accordance with the current access mode;

b) so performing the task thereby identified as, in at least some instances, to find one or more further tasks to be performed; and

c) pushing onto that task queue task identifiers that identify any tasks thus found.

16. (Previously Presented) A method as defined in claim 15 wherein pushing occurs at the bottom end of each provided task queue, popping in accordance with the FIFO access mode occurs at the top end of each provided task queue and popping in accordance with the LIFO access mode occurs at the bottom end of each provided task queue.

17. (Previously Presented) A method as defined in claim 15 wherein queue accesses in each provided task queue are circular.

18.-19. (Canceled.)

20. (Currently Amended) A method as defined in claim ~~19~~ 15 wherein the task identifiers are identifiers of the objects associated with tasks that the task identifiers identify.

21. (Original) A method as defined in claim 20 wherein the task identifiers are pointers to the objects associated with the tasks that the task identifiers identify.

22. (Currently Amended) A method as defined in claim ~~18~~ 15 wherein, in at least some instances, an execution thread associated with a task queue that is empty:

A) pops a task identifier from a task queue other than the one with which it is associated;

B) so performs the task thereby identified as, in at least some instances, to find one or more further tasks to be performed; and

C) pushes onto the task queue associated with it task identifiers that identify any tasks thus found.

23. (Original) A method as defined in claim 22 wherein each said dynamically identified task is the garbage-collection task of performing, for a given object associated with that task, processing that includes identifying in the given object

references to other objects and thereby identifying the tasks of performing similar processing for those other objects.

24. (Currently Amended) A storage medium containing instructions readable by a computer system to cause the computer system to:

A) provide ~~at least one task queue~~ a plurality of task queues, each having a top end and a bottom end and in which can be stored and from which can be retrieved task identifiers, which identify tasks to be performed; and

B) for each provided task queue, employ a separate execution thread associated therewith to:

i) select repeatedly a current access mode from one of a LIFO access mode and a FIFO access mode in accordance with a mode-selection criterion; and

ii) perform dynamically identified tasks, wherein each said dynamically identified task is the garbage-collection task of performing, for a given object associated with that task, processing that includes identifying in the given object references to other objects and thereby identifying the tasks of performing similar processing for those other objects, by repeatedly:

a) popping a task identifier from one of the top end or the bottom end of that task queue in order to access that task queue in a LIFO access mode or a FIFO access mode in accordance with the current access mode ;

b) so performing the task thereby identified as, in at least some instances, to find one or more further tasks to be performed; and

c) pushing onto that task queue task identifiers that identify any tasks thus found.

25. (Previously Presented) A storage medium as defined in claim 24 wherein pushing occurs at the bottom end of each provided queue, popping in

accordance with the FIFO access mode occurs at the top end of each provided queue, and popping in accordance with the LIFO access mode occurs at the bottom end of each provided queue.

26. (Previously Presented) A storage medium as defined in claim 24 wherein queue accesses in each provided task queue are circular.

27.-28. (Canceled.)

29. (Currently Amended) A storage medium as defined in claim ~~28~~ 24 wherein the task identifiers are identifiers of the objects associated with tasks that the task identifiers identify.

30. (Original) A storage medium as defined in claim 29 wherein the task identifiers are pointers to the objects associated with the tasks that the task identifiers identify.

31. (Currently Amended) A storage medium as defined in claim ~~27~~ 24 wherein, in at least some instances, an execution thread associated with a task queue that is empty

A) pops a task identifier from a task queue other than the one with which it is associated;

B) so performs the task thereby identified as, in at least some instances, to find one or more further tasks to be performed; and

C) pushes onto the task queue associated with it task identifiers that identify any tasks thus found.

32. (Original) A storage medium as defined in claim 31 wherein each said dynamically identified task is the garbage-collection task of performing, for a given object associated with that task, processing that includes identifying in the

given object references to other objects and thereby identifying the tasks of performing similar processing for those other objects.

33. (Currently Amended) A ~~signal representing a sequence of instructions that, when they are executed by~~ computer system, ~~cause the computer system to~~ comprising:

A) ~~provide at least one task queue~~ means for providing a plurality of task queues, each having a top end and a bottom end and in which can be stored and from which can be retrieved task identifiers, which identify tasks to be performed; and

B) for each provided task queue, ~~employ~~ means for employing a separate execution thread associated therewith to:

i) select repeatedly a current access mode from one of a LIFO access mode and a FIFO access mode in accordance with a mode-selection criterion; and

ii) perform dynamically identified tasks, wherein each said dynamically identified task is the garbage-collection task of performing, for a given object associated with that task, processing that includes identifying in the given object references to other objects and thereby identifying the tasks of performing similar processing for those other objects, by repeatedly:

a) popping a task identifier from one of the top end or the bottom end of that task queue in order to access that task queue in a LIFO access mode or a FIFO access mode in accordance with the current access mode ;

b) so performing the task thereby identified as, in at least some instances, to find one or more further tasks to be performed; and

c) pushing onto that task queue task identifiers that identify any tasks thus found.

34.-42. (Canceled.)

43. (Previously Presented) A computer system as defined in claim 1 wherein the mode-selection criterion is based on the number of entries in the task queue.

44. (Canceled).

45. (Previously Presented) A method as defined in claim 15 wherein the mode-selection criterion is based on the number of entries in the task queue.

46. (Previously Presented) A storage medium as defined in claim 24 wherein the mode-selection criterion is based on the number of entries in the task queue.

47. (Currently Amended) A ~~signal~~ computer system as defined in claim 33 wherein the mode-selection criterion is based on the number of entries in the task queue.